PATENT ABSTRACTS OF JAPAN

(11)Publication number : 2002-223214
(43)Date of publication of application : 09.08.2002

(51)Int.Cl. H04L 12/28
G06F 9/46
G06F 13/00
G06F 15/16
H04L 29/08

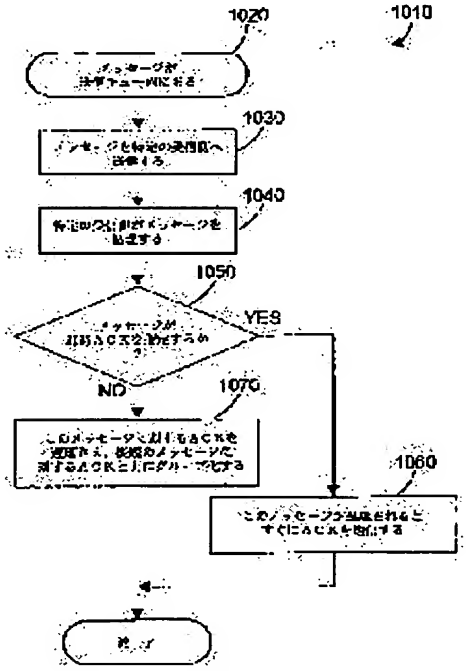
(21)Application number : 2001-336277
(22)Date of filing : 01.11.2001

(71)Applicant : INTERNATL BUSINESS MACH CORP <IBM>
(72)Inventor : BLOCK TIMOTHY R

(30)Priority
Priority number : 2000 718924 Priority date : 22.11.2000 Priority country : US

(54) METHOD AND APPARATUS FOR COMMUNICATION BETWEEN COMPUTER SYSTEMS USING SLIDING TRANSMISSION WINDOW FOR ORDERED MESSAGE IN CLUSTER COMPUTER ENVIRONMENT

(57)Abstract:
PROBLEM TO BE SOLVED: To provide an apparatus and a method by which a sliding transmission window is used for communicating with another computer in a cluster.
SOLUTION: A cluster computer system has a plurality of computer systems (or nodes) which can become members of a group working on a specific task and are connected to each other through one or more networks. Each node includes a cluster communication mechanism incorporating a cluster engine and the sliding transmission window, and one or more service tasks which process messages. The sliding transmission window enables the node to transmit a plurality of messages without waiting for individual acknowledgements to the messages. The sliding transmission window also enables the node to transmit a single confirmation message with respect to a plurality of received messages when the node receives the messages.



LEGAL STATUS

[Date of request for examination] 01.11.2001
[Date of sending the examiner's decision of rejection]
[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]
[Date of final disposal for application]
[Patent number]
[Date of registration]
[Number of appeal against examiner's decision of rejection]
[Date of requesting appeal against examiner's decision of rejection]
[Date of extinction of right]

(19)日本国特許庁（ＪＰ）　(12)公開特許公報（Ａ）　(11)特許出願公開番号
特開2002－223214
（P2002－223214A）
(43)公開日　平成14年８月９日(2002.8.9)

(51)Int.Cl.⁷		識別記号	FI			テーマコード（参考）
H04L	12/28	207	H04L	12/28	207	5B045
G06F	9/46	360	G06F	9/46	360F	5B098
	13/00	520		13/00	520A	5K033
	15/16	640		15/16	640A	5K034
H04L	29/08		H04L	13/00	307Z	

審査請求　有　請求項の数12　ＯＬ　（全 17 頁）

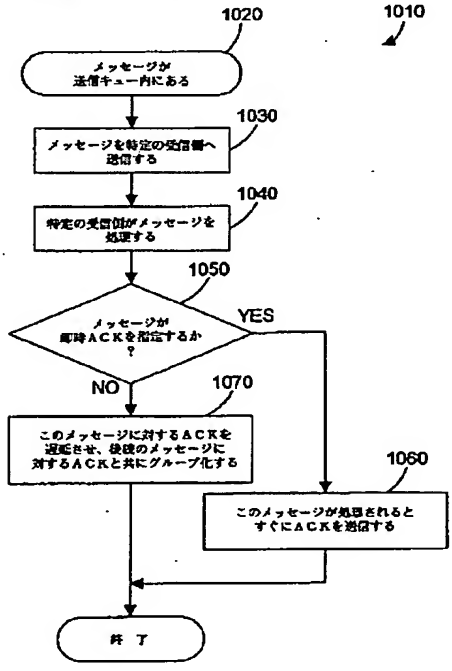
(21)出願番号　特願2001－336277(P2001－336277)
(22)出願日　平成13年11月１日(2001.11.1)
(31)優先権主張番号　０９／７１８９２４
(32)優先日　平成12年11月22日(2000.11.22)
(33)優先権主張国　米国（ＵＳ）

(71)出願人　390009531
インターナショナル・ビジネス・マシーンズ・コーポレーション
INTERNATIONAL BUSINESS MASCHINES CORPORATION
アメリカ合衆国10504、ニューヨーク州　アーモンク　（番地なし）
(74)代理人　100086243
弁理士　坂口　博　（外２名）

最終頁に続く

(54)【発明の名称】　クラスタ・コンピュータ環境において順序付きメッセージのためのスライディング送信ウィンドウを用いてコンピュータ・システム間の通信を行う装置および方法

(57)【要約】
【課題】　クラスタ内の他のコンピュータ・システムと通信するためにスライディング送信ウィンドウを用いる装置および方法を提供する。
【解決手段】　クラスタ・コンピュータ・システムは、特定のタスク上で働くグループのメンバになり得る、１以上のネットワークによって相互に接続された複数のコンピュータ・システム（またはノード）を有する。各ノードは、クラスタ・エンジン，スライディング送信ウィンドウを含むクラスタ通信機構，およびメッセージを処理する１以上のサービス・タスクを含む。スライディング送信ウィンドウは、ノードが、各々のメッセージに対する個々の確認応答を待つことなしに複数のメッセージを発信することを可能にする。スライディング送信ウィンドウは、また、複数のメッセージを受信したノードが、複数の受信メッセージに対して単一の確認メッセージを送信することを可能にする。



【特許請求の範囲】

【請求項１】装置であって、

少なくとも１つのプロセッサ、

前記少なくとも１つのプロセッサに接続されたメモリ、

少なくとも１つの他のコンピュータ・システムに接続されたネットワークへ前記装置を接続するネットワーク・インターフェース、前記メモリに存在し前記少なくとも１つのプロセッサによって実行されるクラスタ通信機構、前記クラスタ通信機構は、次の順序付きメッセージを発信するよりも前に少なくとも１つの他のコンピュータ・システムからの確認メッセージを待つことなしに、前記少なくとも１つの他のコンピュータ・システムへ少なくとも１つの順序付きメッセージを伝達するスライディング送信ウィンドウを有する、

を備える装置。

【請求項２】各順序付きメッセージは、前記順序付きメッセージに対する確認メッセージを遅らせて少なくとも１つの後続の確認メッセージと共にグループ化できるか否か指示する情報を有するヘッダを含む請求項１に記載の装置。

【請求項３】前記確認メッセージは、１から複数の順序付きメッセージを確認する請求項２に記載の装置。

【請求項４】コンピュータ・システムのクラスタを備えるネットワーク・コンピュータ・システムであって、

前記コンピュータ・システムの各々は、

各コンピュータ・システムを、前記クラスタ内の他のコンピュータ・システムへネットワークを介して接続するネットワーク・インターフェース、

メモリ、

前記メモリに存在するクラスタ通信機構、前記クラスタ通信機構は、次の順序付きメッセージを発信するよりも前に少なくとも１つの他のコンピュータ・システムからの確認応答を待つことなしに、前記少なくとも１つの他のコンピュータ・システムへ少なくとも１つの順序付きメッセージを伝達するスライディング送信ウィンドウを有する、

を含むネットワーク・コンピュータ・システム。

【請求項５】各順序付きメッセージは、前記順序付きメッセージに対する確認メッセージを遅らせて少なくとも１つの後続の確認メッセージと共にグループ化できるか否か指示する情報を有するヘッダを含む請求項４に記載のネットワーク・コンピュータ・システム。

【請求項６】クラスタ・コンピュータ環境においてタスクを処理するコンピュータ実装方法であって、前記方法は、

次の順序付きメッセージを発信するよりも前に、順序付きメッセージを受信した前記クラスタ内の各コンピュータ・システムからの確認応答を待つことなしに前記クラスタ内の少なくとも１つの他のコンピュータ・システムへ少なくとも１つの順序付きメッセージを伝達するスライディング送信ウィンドウを有する、前記クラスタ内の第１のコンピュータ・システム上で実行するクラスタ通信機構を与えるステップと、

前記クラスタ通信機構が、前記クラスタ内の少なくとも１つの他のコンピュータ・システムへ第１の順序付きメッセージを送信するステップと、

前記クラスタ通信機構が、前記クラスタ内の前記少なくとも１つの他のコンピュータ・システムからの前記第１の順序付きメッセージに対する応答を待つことなしに、第２の順序付きメッセージを送信するステップとを含む方法。

【請求項７】前記クラスタ内の前記少なくとも１つの他のコンピュータ・システムが、前記第１および第２の順序付きメッセージの双方を確認する単一の確認メッセージを前記クラスタ通信機構へ送信することによって前記第１および第２の順序付きメッセージに応答するステップをさらに含む請求項６に記載の方法。

【請求項８】前記第１および第２の順序付きメッセージは、各々、前記第１および第２の順序付きメッセージに対する確認メッセージを遅らせて少なくとも１つの後続の確認メッセージと共にグループ化できるか否か指示する情報を有するヘッダを含む請求項６に記載の方法。

【請求項９】プログラム・プロダクトであって、

（Ａ）次の順序付きメッセージを発信するよりも前に、少なくとも１つの他のコンピュータ・システムからの確認応答を待つことなしにクラスタ内の少なくとも１つの他のコンピュータ・システムへ少なくとも１つの順序付きメッセージを伝達するスライディング送信ウィンドウを有するクラスタ通信機構（Ａ１）を含むコンピュータ・プログラム、

（Ｂ）前記コンピュータ・プログラムを伝達するコンピュータ読み取り可能信号伝達媒体、

を備えるプログラム・プロダクト。

【請求項１０】前記信号伝達媒体は、記録可能媒体よりなる請求項９に記載のプログラム・プロダクト。

【請求項１１】前記信号伝達媒体は、伝送媒体よりなる請求項９に記載のプログラム・プロダクト。

【請求項１２】各順序付きメッセージは、前記順序付きメッセージに対する確認メッセージを遅らせて少なくとも１つの後続の確認メッセージと共にグループ化できるか否か指示する情報を有するヘッダを含む請求項９に記載のプログラム・プロダクト。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】本発明は、広くデータ処理に関し、特に、ネットワーク上のコンピュータ間のタスクの共用に関する。

【０００２】

【従来の技術】コンピュータ時代の始まり以後、コンピュータ・システムは、工学設計，マシンおよびプロセス

・コントロール，そして情報記憶およびアクセスを含む人間の活動の多くの分野において必要不可欠になった。コンピュータの初期において、銀行のような企業，産業界，および政府機関は、彼らの要求を満たす単一のコンピュータを購入していた。しかし、１９５０年代の初頭までには、多くの企業が複数のコンピュータを有し、１つのコンピュータから他のコンピュータへデータを移す必要性が明白となった。このときから、コンピュータが共働することを可能にするためにコンピュータ・ネットワークが開発され始めた。

【０００３】ネットワーク・コンピュータは、どの単一コンピュータも実行できないタスクを実行することができる。加えて、ネットワークは、低コスト・パーソナル・コンピュータ・システムが、より大規模のシステムに接続して、このような低コスト・システムが単独では実行できないタスクを実行することを可能にする。米国内の大部分の企業は、現在、１以上のコンピュータ・ネットワークを有する。ネットワークのトポロジおよびサイズは、ネットワーク化されるコンピュータ・システムおよびシステム管理者の設計に従って変化し得る。実際に、企業が複数のコンピュータ・ネットワークを有することは非常に一般的なことである。多くの巨大企業は、企業内の大多数のコンピュータを相互に効果的に接続するローカル・エリア・ネットワーク（ＬＡＮ）および広域ネットワーク（ＷＡＮ）の複雑な混合を有する。

【０００４】ネットワーク上で共につながれた複数のコンピュータに関して、ネットワーク・コンピュータを用いて、タスクの異なる部分をネットワーク上の異なるコンピュータに委託し、それらが続いて各自の個々の部分を並列に処理できることによってタスクを完了することができるということが速やかに明白となった。ネットワーク上の共用コンピューティングに対する１つの具体的な構成において、タスクの異なる部分に対して並列に働くことができるネットワーク上のコンピュータ・システムのグループを定義するためにコンピュータ“クラスタ”の概念が用いられてきた。加えて、コンピュータ・クラスタは、きわめて信頼性が高いサービスを伴う“単一システム・イメージ”を提供する。クラスタ内の複数のシステムは、ユーザにとって１つのコンピュータ・システムのように見え得る、そして、ユーザが必要とするサービスおよびリソースは、たとえクラスタ内のコンピュータ・システムのうちの１つが故障し、あるいはメンテナンスのためにダウンされていても常に利用できる。

【０００５】クラスタ内のコンピュータが共働してタスクを実行するための１つの方法は、順序付きメッセージの概念を用いる。順序付きメッセージ・システムにおいて、各々のメッセージは、典型的にＩＰマルチキャストを用いて全てのノードへ伝達され、メッセージの順序は、全てのノードが同じ順序で一定の発信元からのメッセージを認識するように強制される。先行技術クラスタ・コンピュータ環境において、各々のメッセージは、続けて次のメッセージを処理するよりも前に各ノードによって処理される。すなわち、順序付きメッセージを使用するクラスタ・コンピュータ環境において通信を行う先行技術は、単一メッセージを備える送信ウィンドウを有し、これは、１つの固定された送信ウィンドウ・サイズである。

【０００６】

【発明が解決しようとする課題】“スライディング送信ウィンドウ”の概念は、伝送制御プロトコル（ＴＣＰ）二地点間メッセージと関連して技術上既知である。スライディング送信ウィンドウは、複数のメッセージが、次のメッセージを送信するよりも前に各々のメッセージに対する個々の確認応答を待つことなしに送信されることを可能にする。スライディング送信ウィンドウは、ＴＣＰを用いる二地点間通信として知られているが、スライディング送信ウィンドウは、それが現在まで解決されていない特定の問題を生じさせるので、クラスタ・コンピュータ環境において使用されなかった。特に、ＩＰはメッセージの順序を強制していないので、個々のノードに対するマルチキャストである順序付きメッセージを、全てのノード上で同じ順序で処理するという要件は、先行技術ＴＣＰスライディング送信ウィンドウを用いて実現できない。こういう理由で、クラスタ・コンピュータ環境におけるＩＰマルチキャスト通信は、スライディング送信ウィンドウの使用によって利益を得られなかった。クラスタ・コンピュータ環境において使用可能なスライディング送信ウィンドウを提供するメカニズムおよび方法がなければ、クラスタ・コンピュータ・システムの性能は、現行の１つの固定された送信ウィンドウ・サイズによって制限され続けることになる。

【０００７】

【課題を解決するための手段】好適な実施の形態によれば、クラスタ・コンピュータ・システムは、特定のタスク上で働くグループのメンバになり得る、１以上のネットワークによって相互に接続された複数のコンピュータ・システム（またはノード）を含む。各々のノードは、クラスタ・エンジン，スライディング送信ウィンドウを有するクラスタ通信機構，およびメッセージを処理する１以上のサービス・タスクを含む。スライディング送信ウィンドウは、ノードが、各メッセージに対する個々の確認応答を待つことなしに複数のメッセージを発信することを可能にする。スライディング送信ウィンドウは、また、複数のメッセージを受信したノードが、複数の受信メッセージに対して単一の確認メッセージを送信することを可能にする。クラスタ内の他のコンピュータ・システムと通信するためにスライディング送信ウィンドウを用いることにより、クラスタ内の通信トラフィックは著しく削減され、それによってクラスタの全体性能を高める。加えて、同時に送信された複数のメッセージ間の

待ち時間は、非常に効果的に短縮される。

【０００８】本発明の上述および他の特徴および利点は、添付の図面において例示される下記の本発明の好適な実施の形態のより詳細な説明から明らかにすることができる。

【０００９】

【発明の実施の形態】本発明の好適な実施の形態を、同一の指示が同一の要素を示す添付の図面と関連して以下に説明する。

【００１０】本発明は、ネットワーク上で接続されたコンピュータ上でタスクの一部を共用することによって達成される。ネットワーク概念に精通していない読者のために、後述の簡潔な概要は、読者が本発明を理解するのに役立ち得る背景情報を提供する。

【００１１】１．概要

ネットワーク・コンピュータ・システム

コンピュータをネットワーク上で相互に接続することは、ある種のネットワーキング・ソフトウェアを必要とする。長年にわたって、ネットワーキング・ソフトウェアの能力および複雑さは著しく増大した。ネットワーキング・ソフトウェアは、典型的に、ネットワーク上のコンピュータ間で情報を交換するためのプロトコルを定義する。多くの異なるネットワーク・プロトコルが技術上知られている。商業的に入手可能なネットワーキング・ソフトウェアの代表例は、Ｎｏｖｅｌｌ　Ｎｅｔｗａｒｅおよびｗｉｎｄｏｗｓ（Ｒ）　ＮＴであり、各々がコンピュータ間で情報を交換するための異なるプロトコルを実装する。

【００１２】近年非常にポピュラーになった１つの重要なコンピュータ・ネットワークは、インターネットである。インターネットは、コンピュータおよびネットワークの急増から発展し、コンピュータ・システムの複雑なワールドワイド・ネットワークへ進化した。インターネットを用いて、ユーザは、単一のワークステーションから世界中のコンピュータにアクセスできる。ＴＣＰ／ＩＰ（伝送制御プロトコル／インターネット・プロトコル）は、インターネットを介した２つのコンピュータ間の二地点間通信のために今日広く使われているネットワーク・プロトコルの代表例である。加えて、ＴＣＰ／ＩＰの使用は、また、企業内部のさらに多くのローカル・エリア・ネットワーク（ＬＡＮ）およびイントラネットに迅速に広がっている。

【００１３】ユーザ・データグラム・プロトコル（ＵＤＰ）は、既知のネットワーク・プロトコルの他の例である。ＵＤＰは、ＴＣＰと関連したオーバーヘッドを多くは備えないだけでなく、ＴＣＰの信頼性を備えない。ＴＣＰにおいて、２つのコンピュータ・システムは、当該２者の間に“コネクション”を設定することによって二地点間で通信する。受信側のノードが、送信側のノードによって送信されたメッセージを受信することができない場合には、送信側のノードは、受信側のノードがメッセージを確認しなかったということを認識し、メッセージを再送することになる。他方、ＵＤＰは、“コネクション”を扱わず、メッセージの受信を確認するための設計された方法を備えない。その結果、送信側のコンピュータ・システムは、メッセージが受信されたか否かを識別する方法を備えない。ＵＤＰは、コンピュータ・クラスタ内のＩＰマルチキャスト環境において首尾よく用いられてきた、しかし、重要なシステム・レベル・コードに、ＵＤＰによって送信され受信されたメッセージを管理して信頼性が高い通信を保証することを要求する。要するに、ＵＤＰを用いることによってＴＣＰのオーバーヘッドの一部を除去することにより、より低いレベルの実装がシステム・レベル・コードのプログラマにとって利用可能となり、それによってコンピュータ・クラスタのための低レベル通信プロトコルの実装に、より高い柔軟性を与える。

【００１４】コンピュータ・クラスタ

先行技術は、コンピュータ・システムのグループに問題の異なる部分上で働かせることの利点を認めている。コンピュータの“クラスタ”の概念は、発展して、より大きいタスクの一部を共用できるネットワーク・コンピュータの事前定義されたグループを包含した。クラスタの１つの具体的な実装は、クラスタ内のコンピュータ間で通信を行うために順序付きメッセージを用いる。順序付きメッセージ・システムにおいて、各メッセージは、全てのノードに伝達され、メッセージの順序は、全てのノードが同じ順序でメッセージを認識するように強制される。複数のコンピュータへ順序付きメッセージを同時にブロードキャストする１つの既知の方法は、ＩＰマルチキャストを使用する。

【００１５】図１を参照すると、５つのコンピュータ・システム（またはノード）１１０からなる単純クラスタ１００が示される。これらのノード間の接続は、論理接続を表し、物理接続は、クラスタ内のノードが相互に論理的に通信可能である限りは好適な実施の形態の範囲内で変化し得る。クラスタ内部で、共働してタスクを成しとげるノードの論理グループに相当する１以上の“グループ”が定義可能である。グループ内の各ノードは、そのグループの“メンバ”といわれる。図２に示されるように、先行技術クラスタ内の各ノード２１０は、システム・レベル・コード２９０およびカーネル２９２を有するオペレーティング・システムを含む。

【００１６】カーネル２９２は、コンピュータ・システム・ハードウェアと直接対話する低レベル・オペレーティング・システム・コードを表す。一番下の層は、ＩＰ／物理層２８０であり、これは物理的な通信媒体を通じて通信するオペレーティング・システム・ソフトウェアの層である。ＩＰ／物理層２８０の上にＵＤＰ層２７０があり、これはコンピュータ・システム間でメッセージ

を交換するためのネットワーク・プロトコルを提供する。クラスタ・トポロジ・サービス２６２およびクラスタ通信２６０は、ＵＤＰ層２７０の上に存在する。クラスタ・トポロジ・サービス２６２は、クラスタの現行トポロジ・ビューを保持し、必要に応じてクラスタにメンバを追加し、あるいはクラスタからメンバを削除することによってクラスタのトポロジを変更するためのサポートを提供する。クラスタ通信２６０は、クラスタ内の各コンピュータ・システムからの順序付きメッセージの伝送および受信のサポートを提供する機構である。クラスタ通信機構２６０は、単一ソースへのおよび単一ソースからのメッセージの順序を保証する。しかし、それぞれに異なるコンピュータ・システムへのあるいはそれぞれに異なるコンピュータ・システムからのメッセージ間の順序は保証しない。クラスタ・エンジン２５０（ＣＬＵＥとしても知られている）は、クラスタ通信機構２６０によって他のノードからメッセージを受信し、全てのソースからの全てのメッセージの総合的な順序を保証する。ＣＬＵＥ２５０は、クラスタ内のノード間の順序付きメッセージを強制するソフトウェア・プロセスである。ＣＬＵＥ２５０が、グループに向けられたそのメンバからのメッセージを受信するとき、ＣＬＵＥ２５０は、利用可能である場合にＩＰマルチキャストを典型的に使用するクラスタ通信機構を介して、グループの全ての登録済みメンバに対してメッセージを送信する。ＣＬＵＥコードの一部は厳密にはカーネル２９２の一部とみなされ、一方ＣＬＵＥの他の部分は厳密にはシステム・レベル・コード２９０とみなされ、これが、クラスタ・エンジン２５０が各々の一部を含むということが図２に示される理由であるということを留意されたい。

【００１７】クラスタ制御層２４０およびクラスタ・グループ・マネージャ２４２が、ＣＬＵＥ層２５０の上に位置する。クラスタ制御２４０は、ノードに対するクラスタ化の構成および活動化を管理し、典型的に、クラスタ環境の管理に適切である様々なクラスタ初期化およびノード管理オペレーションをサポートする。クラスタ・グループ・マネージャ２４２は、全クラスタのグループ・メンバーシップ状況情報のコピーを同期的に保持する一方、クラスタ・ライブラリ関数２３０は、クラスタに他のサポート・サービスを提供する。クラスタ化ＡＰＩコンポーネント２２０は、ジョブ／アプリケーション２１４（例えば図２に示されるジョブ／ａｐｐ２１４Ａおよびジョブ／ａｐｐ２１４Ｂ）を介して基礎クラスタ化機能に外部インターフェースを提供する。クラスタ・マネージャ２１２は、それによってユーザがクラスタ通信パラメータの変更を開始できるユーザ・インターフェースを提供する。

【００１８】クラスタ通信機構２６０の先行技術実施例が図３に示される。サイズが１つの固定送信ウィンドウ３１０は、送信されるべきメッセージを含む送信キュー３２０，どのメッセージが現在作動されているか指示する現行メッセージ属性３３０およびどのノードが現行メッセージを確認したかに関する情報を含むＡＣＫインジケータ３４０と連係して使用される。

【００１９】２．詳細な説明

本発明の好適な実施の形態による装置および方法は、クラスタ・コンピュータ環境におけるスライディング送信ウィンドウを提供する。スライディング送信ウィンドウは、複数のメッセージが、各メッセージに対する個々の応答を待つことなしに送信されることを可能にする。その代わりに、複数のメッセージの受信を確認する単一の確認メッセージを送信可能である。クラスタ内のコンピュータ・システム間の通信のためのスライディング送信ウィンドウの使用は、ネットワーク・トラフィックに大きな削減をもたらし、それによってクラスタの性能を高める。

【００２０】図４を参照すると、好適な実施の形態に従うクラスタ通信機構４６０は、グループの全てのメンバによる各メッセージの受信を個別に確認することなしに、クラスタ内のグループのメンバに複数のメッセージを送信することを可能にするスライディング送信ウィンドウ４１０を含む。送信キュー３２０は、好ましくは、図３の先行技術実施例におけるのと同一であり、しかしながら、代替の実施もまた可能である。現行メッセージ・キュー４３０および保留ＡＣＫキュー４４０は、先行技術におけるそれらの同等物３３０および３４０と個々に比べて機能が拡張され（すなわち、単一のデータ属性からキューあるいはベクトル（２次元の）・エンティティへ）、複数のメッセージを処理する。

【００２１】図４のスライディング送信ウィンドウ４１０の例が図５において図示される。この例に関して、図５の５１０において示されるように、送信ウィンドウは最初にゼロである（メッセージがない）ということが仮定される。５つのメッセージｍ１～ｍ５は、５２０において示されるように、これらのメッセージのうちのどれに対してもＡＣＫを受信することなしに、順々に発信されるということが仮定される。スライディング送信ウィンドウは、５２０において示されるように、それが幅が５つのメッセージｍ１～ｍ５になるまで、一度に一つのメッセージを拡大する。次に、ｍ１およびｍ２に対するＡＣＫが受信され、これが、５３０において示されるように、スライディング送信ウィンドウを３つのメッセージｍ３～ｍ５の幅へ縮小させるということが仮定される。２つのそれ以外のメッセージｍ６およびｍ７が続いて送信され、これが、スライディング送信ウィンドウのサイズを５つのメッセージｍ３～ｍ７へ拡大させるということが仮定される（５４０）。続いて、これらのメッセージの全てに対するＡＣＫが受信され、これが、５５０において示されるように、スライディング送信ウィンドウをゼロへスライドさせ、５６０において示されるよ

うにゼロのサイズへ縮小させるということが仮定される。図５は、なぜ送信ウィンドウが“スライディング”送信ウィンドウと称されるか説明する。そのサイズは、保留中のメッセージの総数および、確認されたメッセージの数に従って変化する（あるいは上下する）。

【００２２】図６を参照すると、コンピュータ・システム６００は、拡張ＩＢＭ　ｉシリーズ・コンピュータ・システムであり、好適な実施の形態に従って相互にネットワーク化可能な１つの適切なタイプのノード１１０（図１）を表す。当業者は、本発明のメカニズムおよび装置が、他のコンピュータ・システムと共にネットワーク化可能ないかなるコンピュータ・システムに対しても等しく適用されるということを理解できる。図６に示されるように、コンピュータ・システム６００は、主記憶装置６２０，大容量記憶インターフェース６３０，端末インターフェース６４０およびネットワーク・インターフェース６５０に接続されたプロセッサ６１０を含む。これらのシステム・コンポーネントは、システム・バス６６０の使用を通して相互接続される。大容量記憶インターフェース６３０は、大容量記憶装置（例えば、直接アクセス記憶装置６５５）をコンピュータ・システム６００へ接続するために使用される。直接アクセス記憶装置６５５の１つの具体的な例は、フロッピー（Ｒ）・ディスケット６９５へデータを格納し、フロッピー（Ｒ）・ディスケット６９５からデータを読み取ることができるフロッピー（Ｒ）・ディスク・ドライブである。

【００２３】主記憶装置６２０は、データ６２２およびオペレーティング・システム６２４を含む。データ６２２は、コンピュータ・システム６００内のあらゆるプログラムへの入力あるいはあらゆるプログラムからの出力に使えるあらゆるデータを表す。オペレーティング・システム６２４は、ＯＳ／４００として産業界で知られているマルチタスク・オペレーティング・システムである。しかしながら、当業者は、本発明の趣旨および範囲は、どの１つのオペレーティング・システムにも限定されないということを理解できる。オペレーティング・システム６２４は、ＯＳシステム・レベル・コード６９０およびカーネル６９２を含む。システム・レベル・コード６９０は、図２におけるＯＳシステム・レベル・コード２９０と同一または同種であってもよく、あるいは好適な実施の形態の範囲内で完全に異なってもよいということを留意されたい。ＯＳカーネル６９２は、クラスタ内の他のノードと通信するために使用されるスライディング送信ウィンドウ４１０を含むクラスタ通信機構４６０を有する。ＯＳカーネル６９２は、ＩＰマルチキャストによってグループの他のメンバと通信するためにクラスタ通信機構４６０によって使用されるＩＰ／物理層（図２の２８０と同様）の一部であるＩＰマルチキャスト・サポート６２６を付加的に含む。好適な実施の形態は、あらゆる組み合わせにおいてコンピュータ・ネットワーク上の二地点間通信およびマルチキャスト通信の双方に明らかに及ぶということを留意されたい。

【００２４】コンピュータ・システム６００は、コンピュータ・システム６００のプログラムが、まるでそれらが複数のより小容量のストレージ・エンティティ例えば主記憶装置６２０およびＤＡＳＤデバイス６５５に対するアクセスの代わりに、大容量の単一のストレージ・エンティティに対するアクセスのみを有するかのように機能することを可能にする、周知の仮想アドレッシング機構を使用する。したがって、データ６２２およびオペレーティング・システム６２４は、主記憶装置６２０に存在するように示されているが、当業者は、これらのアイテムが、必ずしも同時に主記憶装置６２０に全て完全に含まれる必要はないということを理解できる。用語“メモリ”は、コンピュータ・システム６００の全体の仮想メモリを一般的に指示するためにこの中で使用されるということも留意されたい。

【００２５】プロセッサ６１０は、１以上のマイクロプロセッサおよび／または集積回路から構成可能である。プロセッサ６１０は、主記憶装置６２０に格納されたプログラム命令を実行する。主記憶装置６２０は、プロセッサ６１０がアクセスできるプログラムおよびデータを格納する。コンピュータ・システム６００が始動するとき、プロセッサ６１０は、オペレーティング・システム６２４を構成するプログラム命令を最初に実行する。オペレーティング・システム６２４は、コンピュータ・システム６００のリソースを管理する高性能のプログラムである。これらのリソースの一部は、プロセッサ６１０，主記憶装置６２０，大容量記憶インターフェース６３０，端末インターフェース６４０，ネットワーク・インターフェース６５０およびシステム・バス６６０である。単一のプロセッサおよび単一のシステム・バスのみを含むコンピュータ・システム６００を示したが、当業者は、複数のプロセッサおよび／または複数のバスを有するコンピュータ・システムを用いて本発明を実施可能であるということを理解できる。

【００２６】端末インターフェース６４０は、１以上の端末装置６６５をコンピュータ・システム６００へ直接接続するために使用される。非インテリジェント（すなわち、ダム）端末装置あるいは完全プログラマブル・ワークステーションであり得るこれらの端末装置６６５が使用され、システム・アドミニストレータおよびユーザがコンピュータ・システム６００と通信することを可能にする。しかしながら、端末インターフェース６４０が１以上の端末装置６６５との通信をサポートするために与えられる場合には、ユーザとの全ての必要な対話および他のプロセスは、端末インターフェース６４０によって発生可能であるので、コンピュータ・システム６００は、必ずしも端末装置６６５を必要としないということを留意されたい。

【００２７】ネットワーク・インターフェース６５０は、他のコンピュータ・システムおよび／またはワークステーション（例えば図６の６７５）をネットワーク６７０を越えてコンピュータ・システム６００へ接続するために使用される。ネットワーク６７０は、ネットワーク６７０上のコンピュータ・システム６００および他のコンピュータ・システム間の論理接続を表す。本発明は、ネットワーク接続６７０が今日のアナログおよび／またはディジタル手法を用いて成されるか、それとも将来のネットワーク・メカニズムによって成されるかにかかわらず、たとえどのようにしてコンピュータ・システム６００が他のコンピュータ・システムおよび／またはワークステーションと接続されていても等しく適用され得る。加えて、多くの異なるネットワーク・プロトコルがネットワークを実装するために使用可能である。これらのプロトコルは、コンピュータがネットワーク６７０を越えて通信することを可能にする特殊コンピュータ・プログラムである。ＴＣＰ（伝送制御プロトコル）は適切なネットワーク・プロトコルの一例である。

【００２８】ここで、完全に機能的なコンピュータ・システムの状況において本発明を説明してきた、そして説明し続けることになるが、当業者は、様々な形式のプログラム・プロダクトとして本発明を配布できるということ、および、配布を実際に実行するために使用される信号伝達媒体の特定のタイプにかかわらず本発明を等しく適用できるということを理解し得るということを留意されたい。適切な信号伝達媒体の例は、記録可能なタイプの媒体、例えばフロッピー（Ｒ）・ディスク（例えば図６の６９５）およびＣＤ　ＲＯＭ、そして伝送タイプの媒体例えばディジタルおよびアナログ通信リンクを含む。

【００２９】図７を参照すると、３つのノード６００Ａ（ノードＡ），６００Ｂ（ノードＢ）および６００Ｃ（ノードＣ）が全て、ローカル・エリア・ネットワーク（ＬＡＮ）上で相互に接続されているネットワーク構成例７００が示される。これは、技術上知られているコンピュータ・クラスタの最も一般的なネットワーク構成である。図８は、図７のネットワークに対する先行技術のもとでのネットワーク・トラフィックを説明する。ノードＡ内の送信キュー４２０は、それらが送信キュー４２０へ書き込まれたのと同じ順序でノードＢおよびＣへ送信される必要がある３つの順序付きメッセージを有するということが仮定される。ｍ１が送信キュー４２０に最初に受信され、ｍ２およびｍ３がそれに続くということが仮定される。まず、ステップ８１０において、ノードＡがノードＢにｍ１を伝達する。次に、ステップ８１２において、ノードＡがノードＣにｍ１を伝達する。ノードＡは、ｍ２を送信する前に、確認メッセージ（ここではＡＣＫと称される）がｍ１を受信した各ノードから受信されるまで待機しなければならない。それゆえ、ノードＡは、ステップ８２０においてｍ１に対するＡＣＫがノードＢから受信され、ステップ８２２においてｍ１に対するＡＣＫがノードＣから受信されるまで待機する。グループの全ての他のメンバが対応するＡＣＫを用いてｍ１に応答したので、ノードＡは、ノードＢ（ステップ８３０）およびノードＣ（ステップ８３２）に対してｍ２を発信可能である。ノードＡは、ノードＢ（ステップ８４０）およびノードＣ（ステップ８４２）の双方からＡＣＫが受信されるまで再び待機しなければならない。いったんｍ２に対する全てのＡＣＫが受信されると、ノードＡは、ノードＢ（ステップ８５０）およびノードＣ（ステップ８５２）に対してｍ３を発信できる。ノードＡは、ノードＢ（ステップ８６０）およびノードＣ（ステップ８６２）からＡＣＫが受信されるまで再び待機する。図８は、ノード（例えばノードＡ）は、次のメッセージを発信するよりも前にグループの各メンバからのＡＣＫを待たなければならないということを図示する。これは、メッセージの処理が、受信されたのと同じ順序で実行されるということを保証するために行われる。しかしながら、次のメッセージを発信するよりも前に各メッセージに対するＡＣＫを待つことは、全ての発信メッセージを直列化することによる障害をもたらす。発信メッセージのこの直列化は、次のメッセージへ移るよりも前に各ＡＣＫを待つことによるシステム性能の不利益をもたらす。

【００３０】図７におけるのと同一のネットワーク構成に関して図９を概説することにより、好適な実施の形態の概念を図７および８の例と容易に対比し対照することができる。好適な実施の形態において、ノードが、各個別のメッセージに対する確認信号を待つことなしに複数の順序付きメッセージを発信することを可能にするスライディング送信ウィンドウが用いられる。したがって、図９のノードＡは、ステップ９１０においてノードＢに対してｍ１を発信し、ステップ９１２においてノードＣに対してｍ１を発信する。ノードＡは、続いて、ノードＢおよびＣからのｍ１に対するＡＣＫを待つことなしに、ステップ９２０および９２２においてｍ２を、ステップ９３０および９３２においてｍ３を発信することができる。ノードＢおよびＣの各々は、続いて、図８の先行技術において示されるようなノードＢからの３つの別個のＡＣＫおよびノードＣからの３つの別個のＡＣＫを必要としないで、メッセージｍ１，ｍ２およびｍ３の全てを同時に確認する単一のＡＣＫを送信することができる。好適な実施の形態の利点は、従って２倍である。第１に、ノードＡは、各メッセージに対する各ノードからの個別のＡＣＫを待つことなしにメッセージを発信し続け、それによってクラスタに対するワーク・パイプラインをより完全に保つことができる。第２に、ｍ１，ｍ２およびｍ３を受信したノードは、複数のメッセージを同時に確認する単一のＡＣＫを用いて応答し、それによっ

てクラスタ・コンピュータ環境において必要とされるＡＣＫの数を著しく削減することができる。送信側のノードが、次のメッセージを発信するよりも前に各メッセージに対する個別の確認応答を待つことなしに複数のメッセージを発信することを可能にすることにより、そして受信側が、単一の確認応答を用いて複数のメッセージを確認することを可能にすることにより、好適な実施の形態に従うクラスタ・コンピュータ・システムの性能が著しく増大する。

【００３１】図９の通信は、ノードＡとノードＢおよびＣとの間の二地点間通信として示されるということを留意されたい。しかしながら、ＩＰマルチキャストを用いてノード間で通信を行うことは、同様に、好適な実施の形態の範囲内である。この図において、図９のステップ９１０および９１２は、ＩＰマルチキャストを用いてＢおよびＣの双方に対してｍ１をブロードキャストする単一のステップに併合可能である。同様に、ステップ９２０および９２２は、単一のＩＰマルチキャスト・ステップに置き換え可能であり、ステップ９３０および９３２は、単一のＩＰマルチキャスト・ステップに置き換え可能である。好適な実施の形態は、二地点間通信，マルチキャスト通信，およびこれら２つのあらゆる適切な組み合わせに明らかに及ぶ。

【００３２】図１０を参照すると、方法１０１０は、メッセージがノードの送信キュー内にある（ステップ１０２０）ときに、好適な実施の形態に従う１つの例示的な方法において実行されるステップを示す。メッセージが、特定の受信側へ送信される（ステップ１０３０）。メッセージは、グループ内の全てのノードに対するマルチキャスト・メッセージであってもよく、あるいは、各受信側ノードに直接伝達される二地点間メッセージであってもよい。受信側のノードは、グループ内のノードと別個であってもよく、特定のソースからのメッセージの順序を保存しながら、マルチキャストおよび二地点間通信を混合することを可能とするということを留意されたい。特定の受信側は、続いて、メッセージを処理する（ステップ１０４０）。メッセージが即時応答（またはＡＣＫ）を指定する場合（ステップ１０５０＝ＹＥＳ）には、受信側がメッセージを処理するとすぐに各受信側によってＡＣＫが送信される（ステップ１０６０）。一方、メッセージが即時応答を指定しない場合（ステップ１０５０＝ＮＯ）には、メッセージに対するＡＣＫは遅延され、後続のメッセージに対する１以上のＡＣＫと共にグループ化される（ステップ１０７０）。このように、受信側は、ＡＣＫを一緒にして、複数のメッセージが確認されているということを指定する単一のＡＣＫにグループ化することができる。

【００３３】他のノードに対するブロードキャストである各メッセージは、図１１において例として示されるヘッダ１１００のように、様々な情報を有するヘッダを含む。ヘッダ１１００は、メッセージを送信したクラスタ通信機構のバージョン番号を識別するバージョン・フィールド１１１０，メッセージのタイプを識別するタイプ・フィールド１１１２，メッセージに関する情報を提供する様々なフラグを有するフラグ・フィールド１１１４，およびメッセージの長さを識別する長さフィールド１１１６を含む。ソースＩＤフィールド１１２０は、どのノードがメッセージを送信したか識別し、一方、宛先ＩＤフィールド１１３０は、どのノードがメッセージを受信すべきかを識別する。ソースＩＰフィールド１１４０は、送信側のインターネット・プロトコル（ＩＰ）・アドレスを指定し、一方、宛先ＩＰフィールド１１５０は、宛先ノードのＩＰアドレスを指定する。コネクション番号フィールド１１６０は、２つのノード間あるいはノードとサブネット（例えばそのサブネット上のノードのグループ）間のコネクションに対応する番号を含む。シーケンス番号フィールド１１７０および１１８０は、送信された特定のメッセージに対するシーケンス番号を指示する順次的な番号を指示する。ネクスト・フィールド１１９０は、ｓｅｑ１フィールド１１７０内の値を反映する現在は未使用のフィールドである。

【００３４】図１１のフラグ・フィールド１１１４は、図１２に示される遅延ＡＣＫフラグ１２００を含む。遅延ＡＣＫフラグは、ＡＣＫメッセージが即時に送信される必要があるか否かを指示するために使用される。遅延ＡＣＫフラグがセットされている場合には、受信側は待機し、後の時点においてこのメッセージを含む個々のメッセージに対するＡＣＫを合わせてグループ化することができる。遅延ＡＣＫフラグがクリアされている場合には、受信側は、即時に確認応答を行う必要がある。即時に確認応答を行うとは、メッセージが処理された後にＡＣＫを送信することを単に意味し、このＡＣＫは、つまり、このメッセージに加えてあらかじめ処理された１以上のメッセージに対するグループＡＣＫであり得るということを留意されたい。遅延ＡＣＫフラグの意義は、ＡＣＫがこのメッセージのみに対する単一のＡＣＫであるにせよ、あるいはこのメッセージおよび１以上のより早期のメッセージに対するＡＣＫであるにせよ、送信側が次のメッセージを発信可能となる前に、このメッセージに対するＡＣＫが送信側によって必要とされるということである。

【００３５】図１３を参照すると、ネットワーク構成例が、好適な実施の形態の概念をさらに説明するために示される。この構成において、ノードＡはＬＡＮ１上のノードであり、ノードＢおよびＤはＬＡＮ２上のノードであり、ノードＣはＬＡＮ３上のノードである。ＬＡＮ１，ＬＡＮ２およびＬＡＮ３は、全て広域ネットワークＷＡＮ１によって互いに接続されている。図１３のネットワーク構成は、図７の単純ＬＡＮ構成よりも複雑であり、以下に説明される好適な実施の形態の顕著な特色の

一部を説明するのに役立つ。

【００３６】図１４は、図１３に示されるノードＡの機能の一部を説明する。ノードＡは、送信キュー４２０，ＬＡＮ２に対するシーケンス番号を追跡するオブジェクト１４１０，およびＬＡＮ３に対するシーケンス番号を追跡するオブジェクト１４４０を含む。送信キュー４２０は、４つのメッセージｍ１～ｍ４を含む。この例に関して、ｍ１，ｍ２およびｍ４は、ノードＡ，Ｂ，ＣおよびＤであると定義されるグループＸに基づくメッセージであるということが仮定される。ｍ３は、ノードＢに対する二地点間メッセージであるということも仮定される。図１３のノードＡ，Ｂ，ＣおよびＤ間の対話は図１５に示される。

【００３７】図１５の詳細を説明する前に、図１５の基礎にある概念を説明する必要がある。ノードＡは、メッセージが紛失し再送が必要であるとみなされるよりも前にＡＣＫを受信することが許容される最大時間にセットされているメッセージ・タイマを含むということが仮定される。ノードＡは、また、最終メッセージ宛先レジスタを含み、最終メッセージ宛先を、現行メッセージ宛先あるいは次のメッセージ宛先と比較することができる。ノードＢ，ＣおよびＤは、各々、遅延ＡＣＫタイマを含むということも仮定される。各ノードの遅延ＡＣＫタイマは、必要以上に多くの時間が経過する場合に、ＡＣＫが最終的に送信されることを確保するために使用される。ノードがメッセージを受信するとき、ノードは自身の遅延ＡＣＫタイマを始動させる。ノードがメッセージに確認応答するよりも前に遅延ＡＣＫタイマが終了する場合には、ノードはＡＣＫタイマ終了に応答してメッセージに確認応答することとなる。

【００３８】図１５のステップ１において、ノードＡ内の最終メッセージ宛先レジスタがセットされ、次のメッセージ（ｍ１）が、最終メッセージの保管されている宛先と比較される。この例に関して、最終メッセージ（ｍ１より前の）はグループＸに対するものであったと仮定される。最終メッセージ宛先は、次のメッセージｍ２に対する宛先と同じであるので、遅延ＡＣＫフラグがセットされる。メッセージ・タイマが始動され（ステップ２）、ｍ１が送信される（ステップ３）。図１４を再び参照すると、この例に関して、ＬＡＮ２（１４１０）に対するｓｅｑ１（１４２０）およびｓｅｑ２（１４３０）は、いずれも１であり、ＬＡＮ３（１４４０）に対するｓｅｑ１（１４５０）およびｓｅｑ２（１４６０）は、いずれも５０である。これらの数字、１および５０は、任意であり、異なる値を割当ててＬＡＮ２とＬＡＮ３とに対するシーケンス番号を区別することを可能にする。

【００３９】ｍ１を発信することは、ｍ１ヘッダ（図１１を参照）内にｓｅｑ１＝１およびｓｅｑ２＝１を備え、１（真）にセットされた遅延ＡＣＫフラグを備えてノードＢに対してｍ１を送信することによって行われる。メッセージｍ１は、続いて、同じ方法でノードＤに対して送信される。メッセージｍ１は、続いて、ｓｅｑ１＝５０およびｓｅｑ２＝５０を備え、１にセットされた遅延ＡＣＫフラグを備えてノードＣに対して送信される。ノードＢ，ＣおよびＤの各々がｍ１を受信するとき、それらは、それら各自の遅延ＡＣＫタイマを始動させ（ステップ１′）、それら各自のＣＬＵＥにｍ１を渡す（ステップ２′）。同じ値であるシーケンス番号ｓｅｑ１およびｓｅｑ２の双方を有する全てのメッセージは、これが新しい送信ウィンドウの最初のメッセージであるという信号を受信側ノードへ送り、それが、全ての先のメッセージが送信され、未処理のメッセージが１つもなく確認応答されたということを指示するということを留意されたい。

【００４０】次に、ノードＡは、次のメッセージｍ３の宛先を検査し、それが現行メッセージｍ２の宛先と一致するか否か確かめる。メッセージｍ３は、ノードＡおよびノードＢ間の二地点間メッセージであり、一方、メッセージｍ２は、グループＸ内の全てのノード、すなわちＢ，ＣおよびＤに対するものである。これらのメッセージの宛先は一致しないので、遅延ＡＣＫフラグはゼロ（偽）にセットされ、ｍ２が送信される。ノードＢおよびＤに対するメッセージｍ２は、ｓｅｑ２を２にインクリメントし、スライディング送信ウィンドウが２つのメッセージ、ｍ１およびｍ２に増加したということを指示する。一方、遅延ＡＣＫフラグは、メッセージｍ２においてクリアされ、それは、ノードＢ，ＣおよびＤに、次のメッセージを発信するよりも前にスライディング送信ウィンドウ内の全てのメッセージに確認応答することを要求するということを留意されたい。ノードＢ，ＣおよびＤは、ｍ２に対する遅延ＡＣＫフラグがクリアされていることを確かめ、これは、それら各自に未処理のメッセージに確認応答することを要求する。まず、遅延ＡＣＫタイマがクリアされ（ステップ３′）、ｍ２が各自のＣＬＵＥに渡され（ステップ４′）、どのメッセージが確認応答されているのか指示するＡＣＫメッセージのシーケンス番号を用いることによってｍ１およびｍ２の双方を確認する単一のＡＣＫメッセージが、ノードＢ，ＣおよびＤの各々からリターンされる。このようにして、ノードＢおよびＤはｓｅｑ１＝１およびｓｅｑ２＝２を用いて確認応答し、一方ノードＣは、ｓｅｑ１＝５０およびｓｅｑ２＝５１を用いて確認応答する。全ての未処理のメッセージに対するＡＣＫが受信されたので、この時点で、ノードＡはメッセージの送信を再開することができる。

【００４１】ノードＡは、次に、その送信キューを検査し、ｍ３がこの特定の時点において送信キュー内の最終メッセージであるということを調べる（ステップ６）。これに応答して、ノードＡは、自身のメッセージ・タイ

マをリスタートし、メッセージ宛先をリセットする（ステップ７）。メッセージｍ３が続いて発信される。ｍ３は、ノードＡからノードＢへの二地点間メッセージであるということを留意されたい。シーケンス番号は、３にインクリメントされ、遅延ＡＣＫフラグが真にセットされ、ｍ３が続いて送信される（ステップ８）。これに応答して、ノードＢは自身の遅延ＡＣＫタイマを始動させ（ステップ６′）、ノードＢは自身のＣＬＵＥにｍ３を渡す（ステップ７′）。

【００４２】続いて、メッセージｍ４が送信キュー内に到達すると仮定する。メッセージｍ４の宛先（グループＸ）は、最終メッセージｍ３の宛先（ノードＢ）と一致せず（ステップ９）、したがって続行するよりも前に、ノードＡからノードＢへの先の二地点間メッセージは確認応答される必要がある。ＡＣＫ要求メッセージは、シーケンス番号ｓｅｑ１およびｓｅｑ２をメッセージあるいは確認応答されるべきメッセージの値にセットし、ヌル・メッセージ・フラグを真にセットすることによって送信される（ステップ１０）。ノードは、ヌル・メッセージ・フラグを未処理のメッセージに即時に確認応答すべきコマンドと解釈する。その結果、ノードＢは自身の遅延ＡＣＫタイマをリセットし（ステップ８′）、ノードＢはｍ３に対する要求ＡＣＫを渡す（ステップ９′）。好適な実施の形態は、ノードに全てのメッセージを受信して実行することを強制することとなり、ネットワーク帯域幅とＣＰＵリソースとを使用する全てのメッセージの受信を、全てのノードが要求されるわけではないという点で特有である。かわりに、メッセージは、宛先が変わるときに保留メッセージに対するＡＣＫを強制するヌル・メッセージを用いて、自身の意図された受身側へのみ送信される。

【００４３】この時点において、メッセージ・タイマがリスタートされ、メッセージ宛先がリセットされる（ステップ１１）。続いてメッセージｍ４が発信される（ステップ１２）。ノードＢおよびＤに対するシーケンス番号はいずれも４であり、一方ノードＣに対するシーケンス番号は５２であるということを留意されたい。ノードＢは、これまでのメッセージの全てを確認しているので、ノードＢは、次のメッセージのシーケンス番号が４であると予期する。一方、ノードＤは、ノードＡからノードＢへの二地点間メッセージｍ３を確認しなかったので、ノードＤは、次のメッセージのシーケンス番号が３であると予期することに留意されたい。好適な実施の形態に従うシステムのアーキテクチャは、予期された番号よりも大きいシーケンス番号を受信するノードは、送信側が全ての必要なＡＣＫを確認し、ノードは、それが受信しなかったメッセージを受信するよう要求されていなかったということを指示するｓｅｑ１＝ｓｅｑ２をセットした送信側を単に信用するように定義される。したがって、ノードＤが４のシーケンス番号ｓｅｑ１およびｓｅｑ２を有するメッセージｍ４を確認するとき、ノードＤは、先のメッセージはノードＤに向けられていなかったということを信用し、したがってｍ３の欠落を気にせずにｍ４の処理に移る。

【００４４】ノードＡは、送信すべきメッセージをそれ以上有さないと仮定され、したがって、図１５に示されるように、最終的にノードＢ，ＣおよびＤ上の遅延ＡＣＫタイマは、全て終了し（ステップ１０′）、これらのノードの各々に、適切なシーケンス番号を用いてｍ４に対するＡＣＫを渡させることとなる（ステップ１１′）。続いてメッセージ・タイマがリセットされ、メッセージ宛先がリセットされ（ステップ１３）、続いて、ノードＡは次のメッセージを待つ。

【００４５】図１３～１５の具体例は、実施例の具体的な詳細を含む。この実施例は、好適な実施の形態の顕著な特色の一部を説明するために示されており、限定と解釈されるべきではない。好適な実施の形態は、順序付きメッセージを使用するクラスタ・コンピュータ環境におけるスライディング送信ウィンドウを提供するあらゆるメカニズムおよび方法に明らかに及ぶ。

【００４６】好適な実施の形態と関連して説明される本発明は、先行技術よりも著しく優れた改良をここに提供する。これにより、スライディング送信ウィンドウを、マルチキャスト・メッセージ可能なクラスタ・コンピュータ環境において用いることができ、これは、以前は行うことができなかった。スライディング送信ウィンドウを与えることにより、送信側は、次のメッセージを送信するよりも前にメッセージに対するＡＣＫを待つ必要がない。さらに、受信側は遅れて単一の確認メッセージを用いていくつかのメッセージの確認を送信側へ返すことができる。このように、クラスタはよりビジーなメッセージ処理を維持し、ネットワーク・トラフィックは著しく削減され、それによってシステムの性能が高められる。

【００４７】当業者は、本発明の範囲内で多くの変形が実行可能であるということを理解できる。それゆえ本発明の好適な実施の形態と関連して本発明を詳細に示し説明してきたが、当業者は、本発明の趣旨および範囲から外れることなしに、形態および詳細についてのこれらのおよび他の変更をその中に行うことが可能であるということを理解できる。

【００４８】まとめとして、本発明の構成に関して以下の事項を開示する。

（１）装置であって、少なくとも１つのプロセッサ、前記少なくとも１つのプロセッサに接続されたメモリ、少なくとも１つの他のコンピュータ・システムに接続されたネットワークへ前記装置を接続するネットワーク・インターフェース、前記メモリに存在し前記少なくとも１つのプロセッサによって実行されるクラスタ通信機構、前記クラスタ通信機構は、次の順序付きメッセージを発

信するよりも前に少なくとも１つの他のコンピュータ・システムからの確認メッセージを待つことなしに、前記少なくとも１つの他のコンピュータ・システムへ少なくとも１つの順序付きメッセージを伝達するスライディング送信ウィンドウを有する、を備える装置。

（２）各順序付きメッセージは、前記順序付きメッセージに対する確認メッセージを遅らせて少なくとも１つの後続の確認メッセージと共にグループ化できるか否か指示する情報を有するヘッダを含む上記（１）に記載の装置。

（３）前記確認メッセージは、１から複数の順序付きメッセージを確認する上記（２）に記載の装置。

（４）コンピュータ・システムのクラスタを備えるネットワーク・コンピュータ・システムであって、前記コンピュータ・システムの各々は、各コンピュータ・システムを、前記クラスタ内の他のコンピュータ・システムへネットワークを介して接続するネットワーク・インターフェース、メモリ、前記メモリに存在するクラスタ通信機構、前記クラスタ通信機構は、次の順序付きメッセージを発信するよりも前に少なくとも１つの他のコンピュータ・システムからの確認応答を待つことなしに、前記少なくとも１つの他のコンピュータ・システムへ少なくとも１つの順序付きメッセージを伝達するスライディング送信ウィンドウを有する、を含むネットワーク・コンピュータ・システム。

（５）各順序付きメッセージは、前記順序付きメッセージに対する確認メッセージを遅らせて少なくとも１つの後続の確認メッセージと共にグループ化できるか否か指示する情報を有するヘッダを含む上記（４）に記載のネットワーク・コンピュータ・システム。

（６）クラスタ・コンピュータ環境においてタスクを処理するコンピュータ実装方法であって、前記方法は、次の順序付きメッセージを発信するよりも前に、順序付きメッセージを受信した前記クラスタ内の各コンピュータ・システムからの確認応答を待つことなしに前記クラスタ内の少なくとも１つの他のコンピュータ・システムへ少なくとも１つの順序付きメッセージを伝達するスライディング送信ウィンドウを有する、前記クラスタ内の第１のコンピュータ・システム上で実行するクラスタ通信機構を与えるステップと、前記クラスタ通信機構が、前記クラスタ内の少なくとも１つの他のコンピュータ・システムへ第１の順序付きメッセージを送信するステップと、前記クラスタ通信機構が、前記クラスタ内の前記少なくとも１つの他のコンピュータ・システムからの前記第１の順序付きメッセージに対する応答を待つことなしに、第２の順序付きメッセージを送信するステップとを含む方法。

（７）前記クラスタ内の前記少なくとも１つの他のコンピュータ・システムが、前記第１および第２の順序付きメッセージの双方を確認する単一の確認メッセージを前記クラスタ通信機構へ送信することによって前記第１および第２の順序付きメッセージに応答するステップをさらに含む上記（６）に記載の方法。

（８）前記第１および第２の順序付きメッセージは、各々、前記第１および第２の順序付きメッセージに対する確認メッセージを遅らせて少なくとも１つの後続の確認メッセージと共にグループ化できるか否か指示する情報を有するヘッダを含む上記（６）に記載の方法。

（９）プログラム・プロダクトであって、（Ａ）次の順序付きメッセージを発信するよりも前に、少なくとも１つの他のコンピュータ・システムからの確認応答を待つことなしにクラスタ内の少なくとも１つの他のコンピュータ・システムへ少なくとも１つの順序付きメッセージを伝達するスライディング送信ウィンドウを有するクラスタ通信機構（Ａ１）を含むコンピュータ・プログラム、（Ｂ）前記コンピュータ・プログラムを伝達するコンピュータ読み取り可能信号伝達媒体、を備えるプログラム・プロダクト。

（１０）前記信号伝達媒体は、記録可能媒体よりなる上記（９）に記載のプログラム・プロダクト。

（１１）前記信号伝達媒体は、伝送媒体よりなる上記（９）に記載のプログラム・プロダクト。

（１２）各順序付きメッセージは、前記順序付きメッセージに対する確認メッセージを遅らせて少なくとも１つの後続の確認メッセージと共にグループ化できるか否か指示する情報を有するヘッダを含む上記（９）に記載のプログラム・プロダクト。

【図面の簡単な説明】

【図１】ネットワーク上で相互通信できるコンピュータ・システムのブロック図である。

【図２】クラスタ・コンピュータ環境におけるマルチキャスト通信をサポートする先行技術ノード上で実行されるプログラムのブロック図である。

【図３】図２に示されるクラスタ通信機構２６０のブロック図である。

【図４】好適な実施の形態に従うクラスタ通信機構のブロック図である。

【図５】好適な実施の形態のスライディング送信ウィンドウの背景にある概念を示す図である。

【図６】クラスタ内のノードとして使用できる好適な実施の形態に従うコンピュータ・システムのブロック図である。

【図７】クラスタ内でローカル・エリア・ネットワーク（ＬＡＮ）によって相互接続される３つの異なるコンピュータ・システムを示すブロック図である。

【図８】図７のノード間の先行技術対話を示す図である。

【図９】好適な実施の形態に従う図７のノード間の対話を示す図である。

【図１０】好適な実施の形態に従ってスライディング送

信ウィンドウを実装する方法のフロー図である。

【図１１】好適な実施の形態に従ってメッセージ・ヘッダに含まれる情報を示すブロック図である。

【図１２】遅延ＡＣＫフラグが図１２におけるヘッダのフラグ部分１１１４の一部であることを示すブロック図である。

【図１３】異なるローカル・エリア・ネットワーク（ＬＡＮ）上に位置する４つのノード間のハイブリッド・ネットワーク接続例を示すブロック図である。

【図１４】図１３および１５に示されるノードＡの機能を示すブロック図である。

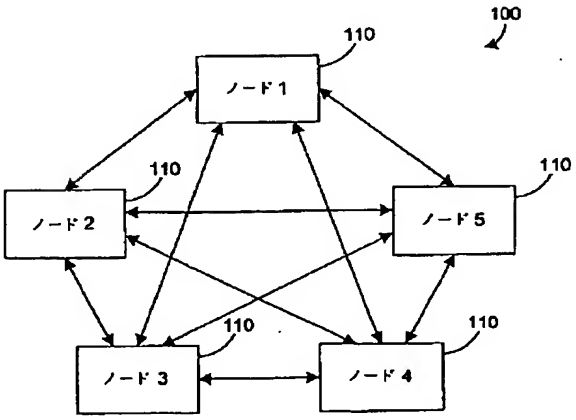
【図１５】好適な実施の形態に従う図１３のノード間の対話を示す図である。

【符号の説明】

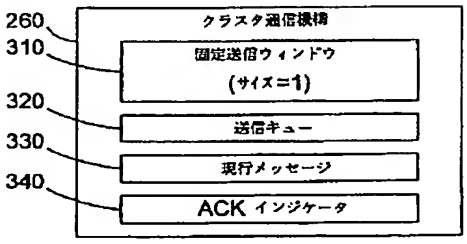
１００　クラスタ
１１０，２１０　ノード
２１２　クラスタ・マネージャ
２１４　ジョブ／アプリケーション
２２０　クラスタ化ＡＰＩコンポーネント
２３０　クラスタ・ライブラリ関数
２４０　クラスタ制御層
２４２　クラスタ・グループ・マネージャ
２５０　クラスタ・エンジン
２６０，４６０　クラスタ通信機構
２６２　クラスタ・トポロジ・サービス
２７０　ＵＤＰ層
２８０　ＩＰ／物理層
２９０，６９０　システム・レベル・コード
２９２，６９２　カーネル
３１０　固定送信ウィンドウ
３２０　送信キュー
３３０　現行メッセージ属性
３４０　ＡＣＫインジケータ

＊４１０　スライディング送信ウィンドウ
４３０　現行メッセージ・キュー
４４０　保留ＡＣＫキュー
６００，６７５　コンピュータ・システム
６００Ａ，６００Ｂ，６００Ｃ　ノード
６１０　プロセッサ
６２０　主記憶装置
６２２　データ
６２４　オペレーティング・システム
６２６　ＩＰマルチキャスト・サポート
６３０　大容量記憶インターフェース
６４０　端末インターフェース
６５０　ネットワーク・インターフェース
６５５　ＤＡＳＤ
６６０　システム・バス
６６５　端末装置
６７０　ネットワーク
６９５　フロッピー（Ｒ）・ディスケット
７００　ネットワーク構成例
１１００　ヘッダ
１１１０　バージョン・フィールド
１１１２　タイプ・フィールド
１１１４　フラグ・フィールド
１１１６　長さフィールド
１１２０　ソースＩＤフィールド
１１３０　宛先ＩＤフィールド
１１４０　ソースＩＰフィールド
１１５０　宛先ＩＰフィールド
１１６０　コネクション番号フィールド
１１７０，１１８０　シーケンス番号フィールド
１１９０　ネクスト・フィールド
１２００　遅延ＡＣＫフラグ
＊　１４１０，１４４０　オブジェクト

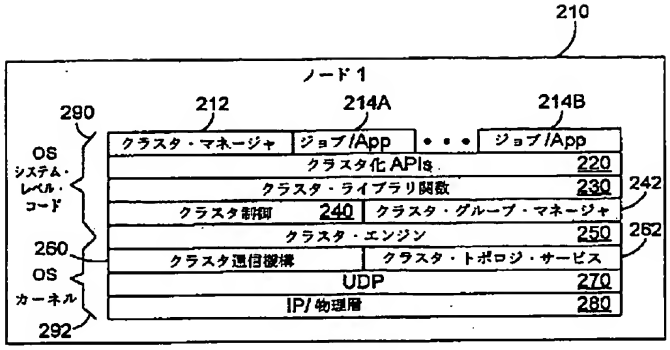
【図１】



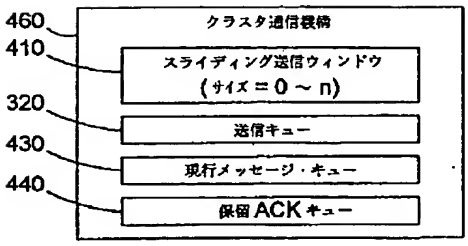
【図３】



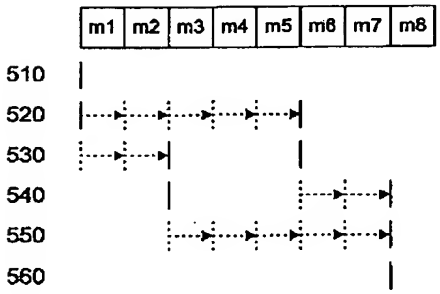
【図２】



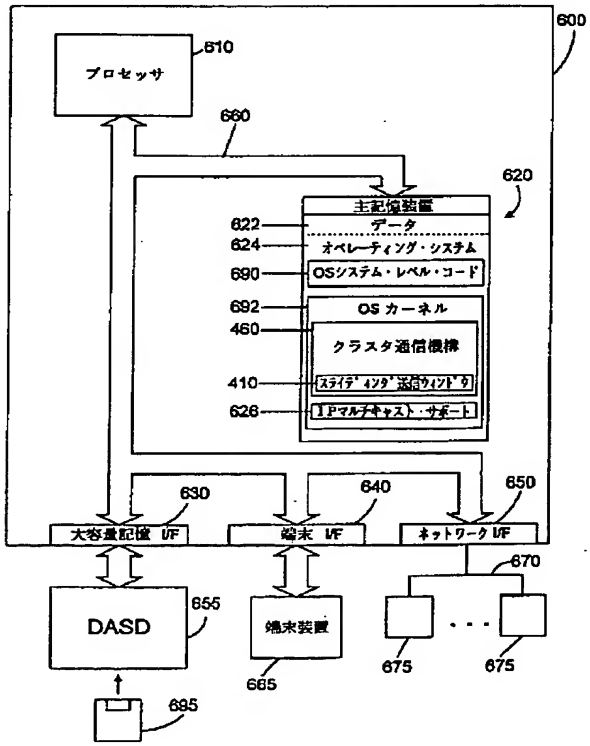
【図４】



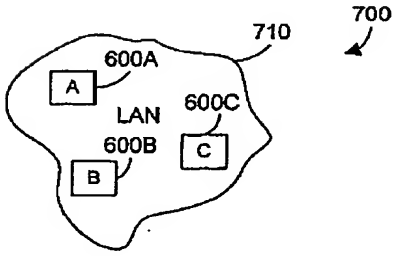
【図５】



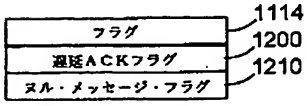
【図６】



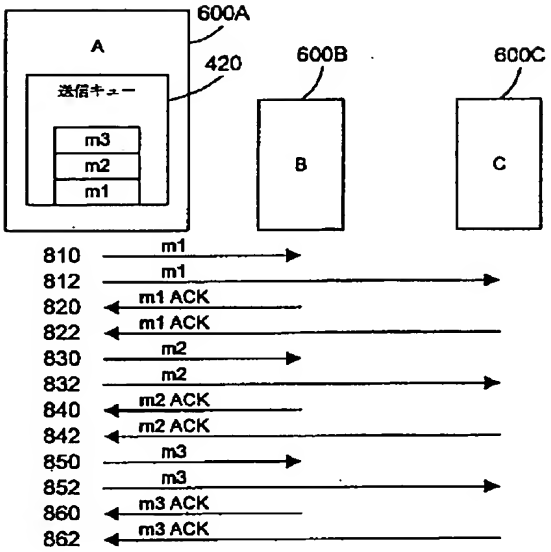
【図７】



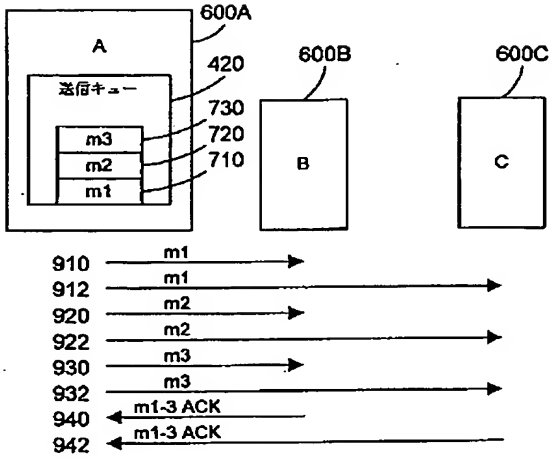
【図１２】



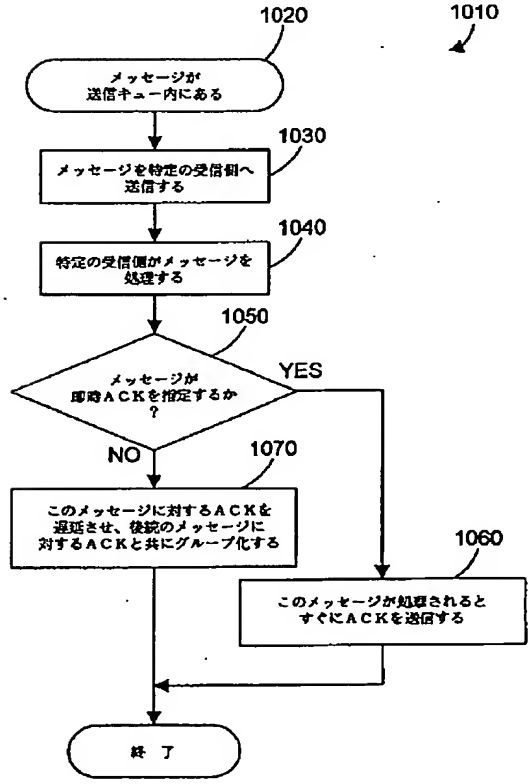
【図８】



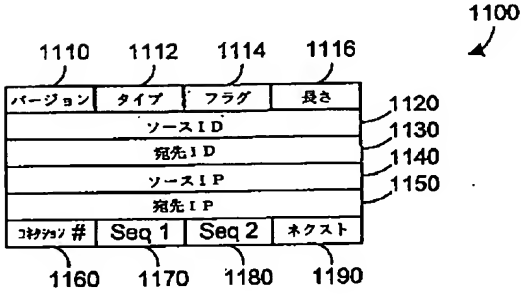
【図９】



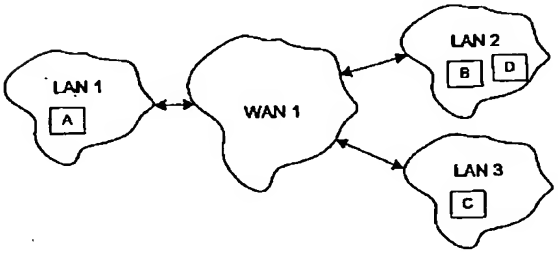
【図１０】



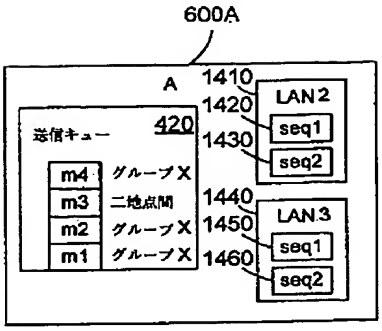
【図１１】



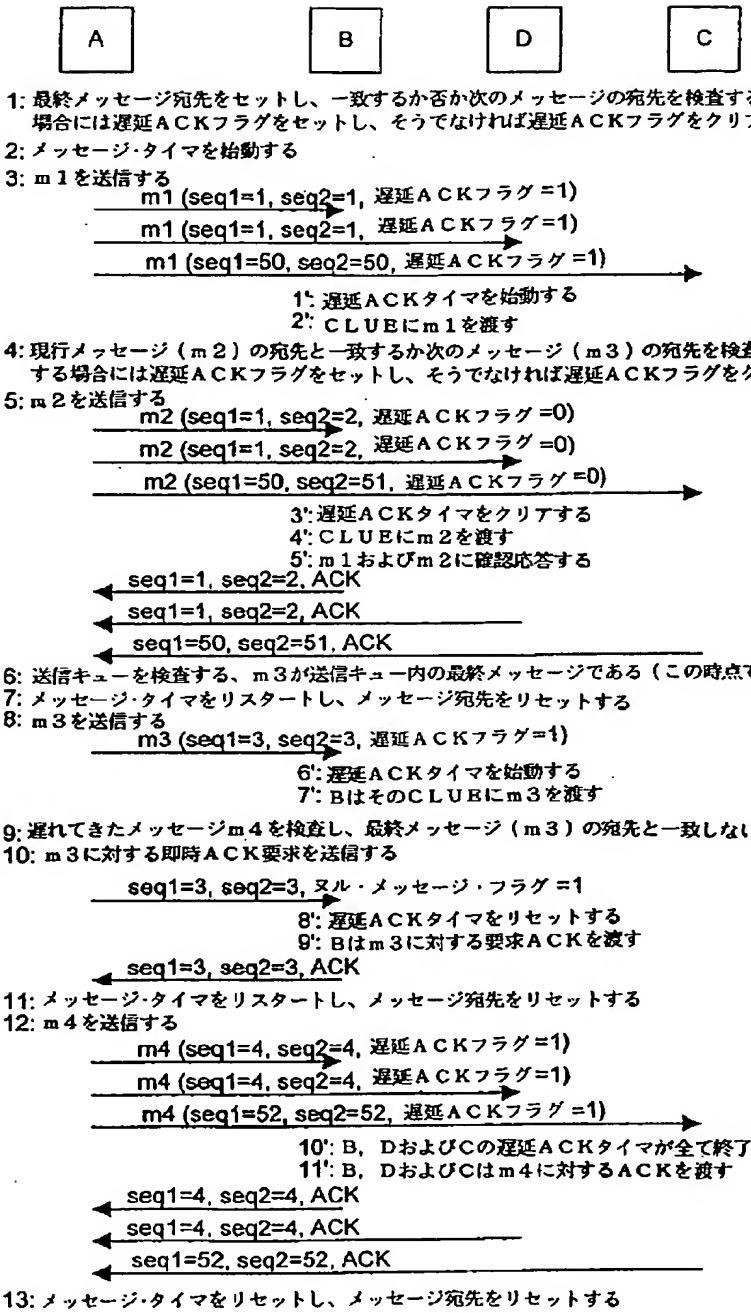
【図１３】



【図１４】


600A
A
1410
1420
LAN2
seq1
seq2
送信キュー
420
1430
m4
グループX
m3
二地点間
1440
LAN3
m2
グループX
1450
seq1
m1
グループX
1460
seq2

【図１５】


A
B
D
C
1: 最終メッセージ宛先をセットし、一致するか否か次のメッセージの宛先を検査する。一致する場合には遅延ＡＣＫフラグをセットし、そうでなければ遅延ＡＣＫフラグをクリアする
2: メッセージ･タイマを始動する
3: ｍ１を送信する
m1 (seq1=1, seq2=1, 遅延ＡＣＫフラグ=1)
m1 (seq1=1, seq2=1, 遅延ＡＣＫフラグ=1)
m1 (seq1=50, seq2=50, 遅延ＡＣＫフラグ=1)
1': 遅延ＡＣＫタイマを始動する
2': ＣＬＵＥにｍ１を渡す
4: 現行メッセージ（ｍ２）の宛先と一致するか次のメッセージ（ｍ３）の宛先を検査する。一致する場合には遅延ＡＣＫフラグをセットし、そうでなければ遅延ＡＣＫフラグをクリアする
5: ｍ２を送信する
m2 (seq1=1, seq2=2, 遅延ＡＣＫフラグ=0)
m2 (seq1=1, seq2=2, 遅延ＡＣＫフラグ=0)
m2 (seq1=50, seq2=51, 遅延ＡＣＫフラグ=0)
3': 遅延ＡＣＫタイマをクリアする
4': ＣＬＵＥにｍ２を渡す
5': ｍ１およびｍ２に確認応答する
seq1=1, seq2=2, ACK
seq1=1, seq2=2, ACK
seq1=50, seq2=51, ACK
6: 送信キューを検査する、ｍ３が送信キュー内の最終メッセージである（この時点で）
7: メッセージ･タイマをリスタートし、メッセージ宛先をリセットする
8: ｍ３を送信する
m3 (seq1=3, seq2=3, 遅延ＡＣＫフラグ=1)
6': 遅延ＡＣＫタイマを始動する
7': ＢはそのＣＬＵＥにｍ３を渡す
9: 遅れてきたメッセージｍ４を検査し、最終メッセージ（ｍ３）の宛先と一致しない
10: ｍ３に対する即時ＡＣＫ要求を送信する
seq1=3, seq2=3, ヌル・メッセージ・フラグ=1
8': 遅延ＡＣＫタイマをリセットする
9': Ｂはｍ３に対する要求ＡＣＫを渡す
seq1=3, seq2=3, ACK
11: メッセージ･タイマをリスタートし、メッセージ宛先をリセットする
12: ｍ４を送信する
m4 (seq1=4, seq2=4, 遅延ＡＣＫフラグ=1)
m4 (seq1=4, seq2=4, 遅延ＡＣＫフラグ=1)
m4 (seq1=52, seq2=52, 遅延ＡＣＫフラグ=1)
10': B, DおよびCの遅延ＡＣＫタイマが全て終了する
11': B, DおよびCはｍ４に対するＡＣＫを渡す
seq1=4, seq2=4, ACK
seq1=4, seq2=4, ACK
seq1=52, seq2=52, ACK
13: メッセージ･タイマをリセットし、メッセージ宛先をリセットする

フロントページの続き

(72)発明者 ティモシー・ロイ・ブロック
アメリカ合衆国 55901 ミネソタ州 ロチェスター エイヴォン レーン エヌダブリュ 4516

Ｆターム(参考) 5B045 BB56 GG01
5B098 AA10 GA04 GD02 GD14
5K033 AA01 AA02 BA04 CA06 CB01 CB04 CB06 CB13 CC01 DB16
5K034 AA02 AA07 EE10 HH01 HH02 HH08 MM18